

**RANCANG BANGUN APLIKASI UNTUK MENTRANSFER  
WARNA DARI CITRA YANG BERWARNA KE CITRA  
GRAYSCALE DENGAN METODE GLOBAL *IMAGE MATCHING***

**TUGAS AKHIR**

Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Sarjana teknik pada jurusan teknik informatika

Oleh :

**MISWAN BUDIANTO**

**10351022927**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU  
PEKANBARU  
2010**

**RANCANG BANGUN APLIKASI UNTUK MENTRANSFER  
WARNA DARI CITRA YANG BERWARNA KE CITRA  
GRAYSCALE DENGAN METODE GLOBAL IMAGE MATCHING**

**MISWAN BUDIANTO**

**10351022927**

Tanggal Sidang: 3 Februari 2010

Periode Wisuda: 25 Februari 2010

Jurusan Teknik Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri Sultan Syarif Kasim Riau

**ABSTRAK**

Citra yang menarik memiliki berbagai macam variasi warna yang sangat indah, selain itu juga banyak citra yang tidak memiliki keindahan warna seperti pada citra *grayscale*. Citra *grayscale* merupakan citra yang hanya menggunakan warna hitam dan putih.

Adapun tujuan dari tugas akhir ini adalah mempelajari dan membuat aplikasi pentransferan warna menggunakan metode *global image matching*, metode ini memiliki beberapa langkah dalam proses pentransferan, langkah pertama yang harus dilakukan adalah proses konversi, langkah kedua yaitu proses histogram, langkah ketiga yaitu mencari rata-rata, langkah keempat yaitu proses pencocokan pixel berfungsi, proses ini merupakan proses terakhir dalam metode ini, langkah-langkah ini harus dilakukan untuk mendapatkan hasil sebuah gambar grayscale yang sudah berwarna.

Dari hasil pengujian menggunakan 55 citra yang berwarna diperoleh tingkat keberhasilan sebesar 90 %. Hasil dari proses pentransferan pada gambar memiliki hasil yang sama meskipun proses pengujian dilakukan lebih dari satu kali.

Kata kunci: *Global image matching* , *Grayscale*, Pengolahan citra digital, Warna.

***DESIGN AND DEVELOP APPLICATION FOR COLOR  
TRANSFERING FROM COLOR IMAGE TO GRAYSCALE IMAGE  
USING GLOBAL IMAGE MATCHING METHOD***

**MISWAN BUDIANTO**

**10351022927**

*Date of Final Exam: Februari 03th, 2010*

*Graduation Ceremony Priod: 25 Februari 2010*

*Engineering Departement of Informatic Technology*

*Faculty of Sciences and Technology*

*State Islamic University of Sultan Syarif Kasim Riau*

**ABSTRACT**

*Beautiful image has a wide variety of beautiful colors, while also many images that do not have the beauty of color as the grayscale image. Grayscale image is an image that only uses black and white.*

*The purpose of this final project is to learn and make the color transfer applications using the global image matching methods, this method has several steps in the process of transferring, the first step is to do the conversion process, the second step of the histogram, the third step is to find the average , the fourth step of pixel matching process works, this process is the last process in this method, these steps should be done to get the results of a grayscale image that is colored..*

*From the results of tests using a 55 color image obtained success rate of 90%. The results of the image transfer process has the same result although the test process more than once.*

*Keywords: Color, Image processing, Global image matching, Grayscale.*

## DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN .....	iii
LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL.....	iv
LEMBAR PERNYATAAN .....	v
LEMBAR PERSEMBAHAN .....	vi
ABSTRAK.....	vii
ABSTRACT .....	viii
KATA PENNGANTAR .....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xiv
DAFTAR TABEL.....	xv
DAFTAR LAMPIRAN.....	xvi
BAB I    PENDAHULUAN .....	I-1
1.1 Latar Belakang .....	I-1
1.2 Rumusan Masalah .....	I-2
1.3 Batasan Masalah.....	I-2
1.4 Tujuan .....	I-2
1.5 Sistematika Penulisan .....	I-3
BAB II   LANDASAN TEORI.....	II-1
2.1 Cita Digital .....	II-1
2.2 Pengolahan Citra .....	II-2
2.3 Grayscale.....	II-5
2.4 Histogram Citra.....	II-7

2.5 Teori dan Aplikasi Statistika dalam gambar .....	II-7
2.5.1 Rata-rata .....	II-8
2.5.2 Simpangan Baku .....	II-8
2.6 Teori Representasi Grafik .....	II-8
2.7 Metode Global Image Matching .....	II-9
 BAB III METODOLOGI PENELITIAN.....	III-1
3.1 Identifikasi Masalah .....	III-2
3.2 Perumusan Masalah .....	III-2
3.3 Studi Pustaka .....	III-2
3.4 Analisis Kebutuhan Aplikasi .....	III-2
3.5 Perancangan Sistem .....	III-3
3.6 Pengujian dan Penerapan .....	III-4
3.7 Kesimpulan dan Saran.....	III-4
 BAB IV ANALISIS DAN PERANCANGAN .....	IV-1
4.1 Hasil analisis .....	IV-1
4.1.1 Analisis kebutuhan sistem.....	IV-1
4.1.1.1 Kebutuhan Masukan .....	IV-1
4.1.1.2 Kebutuhan Fungsi atau Proses .....	IV-1
4.1.1.3 Kebutuhan Keluaran .....	IV-7
4.1.1.4 Kebutuhan Perangkat Lunak.....	IV-8
4.2 Perancangan Sistem .....	IV-8
4.2.1 Deskripsi Umum Sistem .....	IV-8
4.2.2 Perancangan Antar muka .....	IV-9
4.2.2.1 Tampilan Form Utama .....	IV-9
4.2.2.2 Tampilan Form Histogram Gambar Berwarna .....	IV-10
4.2.2.3 Tampilan Form Histogram Gambar Grayscale .....	IV-10
4.2.2.4 Tampilan Hasil Pentransferan .....	IV-11

BAB V IMPLEMENTASI DAN PENGUJIAN .....	V-1
5.1 Implementasi Sistem .....	V-1
5.1.1 Lingkungan Implementasi.....	V-1
5.1.2 Implementasi Pentransferan warna .....	V-2
5.1.2.1 Tampilan Menu Utama .....	V-2
5.2 Pengujian Sistem.....	V-3
5.2.1 Lingkungan Pengujian Sistem.....	V-3
5.2.2 Pengujian.....	V-3
5.2.3 Hasil Pengujian .....	V-3
5.2.4 Kesimpulan Pengujian .....	V-3
 BAB VI PENUTUP .....	 VI-1
6.1 Kesimpulan .....	VI-1
6.2 Saran.....	VI-1
 DAFTAR PUSTAKA	
LAMPIRAN	
DAFTAR RIWAYAT HIDUP	

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Grafika komputer merupakan fasilitas yang sangat baik untuk penyampaian informasi dalam segala bidang. Selain gambar, dalam mode grafik kata-kata pun masih dapat ditampilkan, dan bahkan dapat tampak jauh lebih baik daripada dalam mode teks. Dengan demikian, jelas bahwa penggunaan grafika komputer sangat efektif dalam interaksi manusia dengan komputer.

Citra yang menarik memiliki berbagai macam variasi warna yang sangat indah, selain itu juga banyak citra yang tidak memiliki keindahan warna seperti pada citra *grayscale*. Citra *grayscale* merupakan citra yang hanya menggunakan warna hitam dan putih.

Untuk memperindah koleksi foto-foto hitam putih dan foto-foto yang sudah usang dengan nuansa warna baru yang bisa memperindah citra tersebut, sehingga kita bisa bernostalgia kembali dengan masa lalu kita, dengan cara mentransfer warna yaitu sebuah citra berwarna, ke citra *grayscale* dan menghasilkan citra tujuan yang berwarna.

Pemrosesan terhadap citra dilakukan dengan menggunakan metode *Global Image Matching*. Metode *Global Image Matching* merupakan metode pencocokan warna pada suatu citra digital, yang digunakan untuk mentransfer warna dari citra yang berwarna ke citra *grayscale* dengan cara mencocokkan tingkat kecerahan dan informasi tekstur diantara kedua buah gambar. Didalam metode ini memiliki beberapa langkah-langkah yang harus dilakukan dalam proses pentransferan, langkah pertama yang harus dilakukan adalah proses pengkonversian, langkah kedua proses histogram, langkah ketiga proses rata-rata, langkah keempat proses pencocokan pixel dan yang terakhir adalah proses pentransferan, langkah-langkah

tersebut harus dilakukan untuk mendapatkan hasil sebuah gambar *grayscale* yang sudah ditransfer warna dari citra berwarna.

## **1.2 Rumusan Masalah**

Sebagaimana telah dipaparkan sebelumnya pada latar belakang, maka didapatkan rumusan masalah dari tugas akhir ini yaitu bagaimana merancang dan membangun suatu aplikasi untuk mentransfer warna dari citra berwarna ke citra *grayscale* dengan metode global *image matching*.

## **1.3 Batasan Masalah**

Adapun batasan masalah dalam tugas akhir ini adalah sebagai berikut :

1. Gambar yang berekstensi Bitmap, Tiff dan JPG
2. Gambar yang memiliki ukuran 100x100 pixel.

## **1.4 Tujuan**

Tujuan yang ingin dicapai dalam penyusunan tugas akhir ini adalah :

1. Menganalisa metode global *image matching* dalam pemrosesan pentransferan warna pada citra.
2. Membuat aplikasi untuk mentransfer warna dari citra yang berwarna ke citra *grayscale* menggunakan metode *global image matching*.



## **1.5 Sistematika Penulisan**

Sistematika penulisan dalam penyusunan laporan tugas akhir ini adalah sebagai berikut:

### **BAB I PENDAHULUAN**

Bab ini menjelaskan dasar-dasar dari penulisan laporan tugas akhir, yang terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan, serta sistematika penulisan laporan tugas akhir.

### **BAB II LANDASAN TEORI**

Bab ini membahas teori-teori yang berhubungan dengan topik penelitian, yang terdiri dari Pengolahan Citra Digital dan Metode yang dipakai, yaitu *Metode Goba Image Matching*.

### **BAB III METODOLOGI PENELITIAN**

Bab ini membahas tentang metodologi yang digunakan dalam penelitian dan pengembangan perangkat lunak.

### **BAB IV ANALISIS DAN PERANCANGAN**

Bab ini membahas tentang hasil analisis, deskripsi sistem, fungsi produk, karakteristik pengguna, deskripsi umum kebutuhan, deskripsi perancangan rinci dan perancangan antar muka sistem.

### **BAB V IMPLEMENTASI DAN PENGUJIAN**

Bab ini membahas implementasi dan pengujian yang dilakukan terhadap Aplikasi pentransferan warna ini dengan menggunakan *Metode Global Image Matching*.

### **BAB VI PENUTUP**

Bab ini berisi kesimpulan yang dihasilkan dari pembahasan tentang Aplikasi Aplikasi pentransferan warna ini dengan menggunakan *Metode Global Image Matching* dan beberapa saran sebagai hasil akhir dari penelitian yang telah dilakukan.

## BAB II

### LANDASAN TEORI

#### 2.1. Citra Digital

Citra (*image*) adalah gambar pada bidang dwimatra (dua dimensi). Suatu citra merupakan fungsi intensitas 2 dimensi  $f(x,y)$ , dimana  $x$  dan  $y$  adalah koordinat spasial dan  $f$  pada titik  $(x,y)$  merupakan tingkat kecerahan (*brightness*) citra pada suatu titik. Suatu citra diperoleh dari hasil penangkapan kekuatan sinar atau cahaya yang dipantulkan oleh suatu objek. (Suhendra, 2008). Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, *scanner*, dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam. Bila dikaitkan dengan fungsi matematika, citra dapat direpresentasikan dengan fungsi, yaitu :  $f(x,y)$ ,  $f(x_1,y_2)$ ,  $f(\underline{x})$ , dan seterusnya. (Tharom, 2000)



Gambar 2.1 Contoh Citra (*Image*)

Citra digital atau *digital image* merupakan obyek nyata yang direpresentasikan secara elektronis. Citra digital terdiri dari sejumlah elemen berhingga yang masing-masing mempunyai lokasi dan nilai. Unsur utama citra

digital adalah grid-grid yang berisi elemen obyek yang sangat dasar yang disebut dengan *picture element (pixel)*. Setiap *pixel* mempunyai lokasi dan tingkatan nilai tertentu, sehingga menghasilkan representasi data yang ditangkap oleh mata manusia sebagai bentuk tingkatan warna hitam, putih, abu-abu, hingga penuh dengan warna.

Setiap *bit* dalam *pixel* akan disimpan dalam urutan tertentu oleh komputer dengan perhitungan matematis agar menghasilkan *file* yang optimal yang dibaca oleh perangkat yang mendukungnya. Setiap informasi *bit digital* akan diinterpretasikan dan dibaca oleh komputer agar menghasilkan versi analog untuk ditampilkan dan dicetak oleh media lain. Citra yang dimaksud adalah “citra diam” (*still images*). Citra diam adalah citra tunggal yang tidak bergerak. Sedangkan citra bergerak (*moving images*) adalah rangkaian citra diam yang ditampilkan secara beruntun (sekuensial) sehingga memberi kesan pada mata kita sebagai gambar yang bergerak. Setiap citra di dalam rangkaian itu disebut *frame*. Contohnya terdapat pada gambar-gambar yang tampak pada film layar lebar atau televisi pada hakikatnya terdiri atas ratusan sampai ribuan *frame*. (Rinaldi, 2008)

## **2.2. Pengolahan Citra**

Pengolahan citra atau *image processing* adalah suatu metode yang digunakan untuk memproses, menganalisa, memanipulasi suatu *image* sehingga mendapatkan suatu *image* baru lebih jelas. Dasar pemrosesan suatu obyek dengan menggunakan *image processing* diambil dari kemampuan indera penglihatan manusia yang selanjutnya dihubungkan dengan kemampuan otak manusia.

### **2.2.1. Sejarah Pengolahan Citra Digital**

Minat terhadap bidang pengolahan citra secara digital dimulai pada awal tahun 1921, yaitu pertama kalinya sebuah foto berhasil ditransmisikan secara digital melalui kabel laut dari kota New York ke kota London (*Bartlane Cable Picture Transmission System*). (Wijaya, 2007)

Keuntungan utama yang dirasakan pada waktu itu adalah pengurangan waktu pengiriman foto dari sekitar 1 minggu menjadi kurang dari 3 jam. Foto tersebut dikirim dalam bentuk kode digital dan kemudian diubah kembali oleh *printer telegraph*.

Sekitar tahun 1960 baru tercatat suatu perkembangan pesat seiring dengan munculnya teknologi komputer yang sanggup memenuhi suatu kecepatan proses dan kapasitas memori yang dibutuhkan oleh berbagai algoritma pengolahan citra. Sejak itu, berbagai aplikasi mulai dikembangkan, yang secara umum dapat dikelompokkan ke dalam dua kegiatan :

1. Memperbaiki kualitas suatu gambar (citra) sehingga dapat lebih mudah diinterpretasikan oleh mata manusia.
2. Mengolah informasi yang terdapat pada gambar (citra) untuk keperluan pengenalan obyek secara otomatis oleh suatu mesin.

### **2.2.2. Pengertian Pengolahan Citra Digital**

Pengolahan citra digital atau *digital image processing* sering dikaitkan dengan pemrosesan gambar dua dimensi oleh *computer digital* atau pemrosesan digital data dua dimensi. (Tharom, 2000). Pengolahan citra digital adalah proses pengolahan citra digital dengan alat bantu komputer. (Suhendra, 2008)

Secara obyektif, *image processing* adalah mentransformasikan atau menganalisis suatu gambar sehingga informasi baru tentang gambar dibuat lebih jelas. Umumnya, operasi-operasi pada pengolahan citra diterapkan pada citra bila:

1. Perbaikan atau memodifikasi citra perlu dilakukan untuk meningkatkan kualitas penampakan atau untuk menonjolkan beberapa aspek informasi yang terkandung di dalam citra.
2. Elemen di dalam citra perlu dikelompokkan, dicocokkan, atau diukur.
3. Sebagian citra perlu digabung dengan bagian citra yang lain.

## 2.2. Teori Representasi Warna

Merepresentasikan warna dengan tepat merupakan hal yang penting. Banyak perusahaan saat ini tengah berlomba untuk menyelesaikan masalah ini, karena setiap orang dapat merepresentasikan warna dengan caranya masing-masing. Untuk itulah, diperlukan suatu sistem standar untuk merepresentasikan warna yang biasa disebut *color space*.

Berbicara tentang *color space*, ada 2 jenis *color space* yang sering digunakan dalam aplikasi, yaitu *linear color spaces* dan *non-linear color spaces*.

### 2.2.1 Linear Color Spaces

*Linear color spaces* menyatakan bahwa suatu warna itu terdiri dari 3 warna utama cahaya atau 3 macam panjang gelombang, yaitu LMS (*Long, Middle, Short*). Untuk memperoleh suatu warna tertentu, diperlukan pencocokan warna dengan mengubah komposisi ketiga warna utama tadi. Teknik ini juga digunakan oleh perusahaan cat. Mereka mencampur warna untuk memperoleh suatu warna tertentu. Teknik ini sering digunakan karena lebih sederhana, warna-warna tersebut hanya dicampur secara linier.

Dengan menggunakan teknik ini, terdapat beberapa standar *color spaces* yang menerapkan cara-cara yang berbeda untuk memperoleh warna yang diinginkan, yaitu:

#### 1. CIE XYZ Color Space.

Diciptakan oleh *Commission Internationale de Éclairage* (CIE) pada tahun 1931, *CIE XYZ* merupakan salah satu standar yang cukup terkenal, tetapi sudah kuno. Kelemahan *color space* ini adalah sulitnya untuk mengatur brightness (Gareth Loy: 2002, 1-28).

Untuk merepresentasikan warna, standar ini menggunakan kombinasi penambahan nilai X, Y, dan Z. Ketiga nilai ini selalu bernilai positif, dan diubah menjadi nilai RGB melalui matriks transformasi berikut:

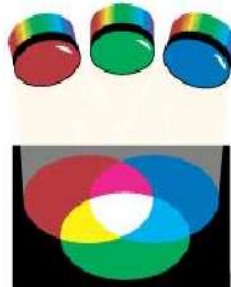
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} \dots\dots\dots (2.1)$$

## 2. RGB Color Space.

Sebagian besar spektrum yang terlihat oleh mata manusia, dapat direpresentasikan dengan menggabungkan warna cahaya merah, hijau, dan biru (RGB) dalam intensitas dan perbandingan yang beragam. Namun, tidak semua warna yang terlihat manusia dapat direpresentasikan dengan kombinasi ketiga nilai warna ini (Gareth Loy: 2002, 1-28).

Perpotongan ketiga warna tersebut menghasilkan warna cyan, magenta, kuning, dan putih. Karena warna RGB dapat dikombinasikan untuk menghasilkan warna putih, maka warna ini disebut juga *additive colors*.

Pada umumnya, nilai RGB berkisar antara 0-255. Kombinasi nilai yang beragam akan menghasilkan warna yang beragam pula. Jika semua nilai RGB bernilai 0, maka akan dihasilkan warna hitam. Sebaliknya, jika semua nilai RGB bernilai 255, maka akan dihasilkan warna putih.



**Gambar 2.2. Additive Colors RGB**

## 2.3. Grayscale.

Proses awal yang banyak dilakukan dalam *image processing* adalah mengubah citra berwarna menjadi citra *grayscale*. Hal ini digunakan untuk menyederhanakan model citra. Citra berwarna terdiri dari 3 *layer* matrik yaitu *R-layer*, *G-layer* dan *B-layer*. Sehingga untuk melakukan proses-proses selanjutnya tetap memperhatikan tiga *layer* tersebut. Bila setiap proses perhitungan dilakukan menggunakan tiga *layer*, berarti dilakukan tiga perhitungan yang sama. Sehingga konsep itu diubah dengan mengubah 3 *layer* di atas menjadi 1 *layer* matrik

*grayscale* dan hasilnya adalah citra *grayscale*. Dalam citra ini tidak ada lagi warna, yang ada adalah derajat keabuan. Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing r, g dan b menjadi citra *grayscale* dengan nilai s, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai r, g dan b sehingga dapat dituliskan menjadi :

$$\text{Gray} = \text{R} + \text{G} + \text{B} / 3 \quad \dots\dots\dots (2.2)$$

Perubahan image dari format RGB menjadi format *grayscale* juga dapat dilakukan dengan menggunakan metode *illuminance grayscale* yang direpresentasikan dalam persamaan berikut :

$$\text{Gray} = 0,299\text{R} + 0,587\text{G} + 0,114\text{B} \quad \dots\dots\dots (2.3)$$

Pada mode ini memanfaatkan warna *gray* (abu-abu) sebanyak 256 tingkat gradasi. Setiap pixel dari gambar *grayscale* mempunyai nilai *brightness* (kecerahan) antara 0 (hitam) hingga 255 (putih). Nilai *grayscale* dapat juga diartikan seperti suatu tinta yang berwarna hitam, dimana mempunyai tingkat kehitaman yang bervariasi (0% sama dengan putih, sedangkan 100% adalah hitam).



**Gambar 2.3** Contoh *image grayscale*

## **2.4. *Histogram Citra***

Citra dengan distribusi intensitas pixel yang tidak merata, dapat diperbaiki dengan suatu proses yang disebut dengan perataan histogram. Perataan histogram ini bertujuan untuk memperoleh histogram yang merata untuk setiap intensitas pixel-nya (Earl Gose: 1996, 275-282). Dengan demikian, dihasilkan citra baru yang memiliki histogram yang lebih terdistribusi, yaitu frekuensi setiap intensitas pixel yang ditempatkan pada sumbu y, kurang lebih sama. Perataan histogram tidak akan benar-benar meratakan histogram. Proses ini hanya akan mendistribusikan ulang distribusi intensitas pixel citra yang bersangkutan. Jika histogram suatu citra memiliki banyak bukit dan lembah, maka setelah proses perataan histogram, histogram tersebut akan tetap memiliki banyak bukit dan lembah, tetapi bukit dan lembah tersebut akan bergeser.

## **2.5. Teori dan Aplikasi Statistika dalam Gambar**

Populasi merupakan keseluruhan pengamatan yang ingin diteliti. Banyaknya pengamatan dalam populasi disebut ukuran populasi, dan disimbolkan dengan  $N$ . Sifat-sifat atau ciri-ciri yang diamati dalam suatu populasi disebut karakter populasi. Sedangkan hasil pengukuran karakteristik dari populasi disebut parameter populasi. Cara mengumpulkan data dari populasi disebut sensus. Dengan sensus, akan diperoleh data yang sebenarnya, artinya data yang dikumpulkan dari populasi merupakan data yang sebenarnya. Tetapi cara sensus ini memiliki kelemahan, yaitu jika jumlah populasi terlalu besar, maka diperlukan waktu, tenaga, dan biaya yang sangat besar pula. Karena kelemahan itu, kegiatan statistik seperti pengumpulan data seringkali dilakukan dengan menggunakan sampel, yaitu sebagian dari obyek populasi itu sendiri. Karakteristik yang dihitung dari sampel disebut statistik. Sedangkan cara mengumpulkan data dari sampel disebut sampling. Berikut di bawah ini, akan dibahas operasi statistika yang sering dipakai dalam aplikasi yang melibatkan gambar, yaitu rata-rata (mean) dan simpangan baku (standar deviasi).



### 2.5.1. Rata-rata ( $\mu$ )

Di dalam gambar, rata-rata nilai kelabu pixel menyatakan tingkat kecerahan gambar tersebut (Earl Gose: 1996, 271-273). Jika terdapat suatu gambar dengan ukuran  $M \times N$  pixel dan nilai pixel  $g$ , maka rata-rata nilai pixel gambar tersebut dapat diketahui melalui rumus di bawah ini:

$$\mu = \frac{1}{MN} \sum_{X=1}^M \sum_{Y=1}^N g(x, y) \quad \dots\dots\dots (2.4)$$

### 2.5.2. Simpangan Baku ( $\sigma$ )

Kontras suatu gambar dapat diketahui melalui besarnya variasi nilai kelabu pixel gambar tersebut (Earl Gose: 1996, 271-273). Salah satu cara untuk mengetahui besarnya variasi ini adalah dengan menghitung akar dari jumlah kuadrat selisih nilai kelabu pixel yang bersangkutan dengan rata-ratanya. Bilangan ini biasa dikenal dengan istilah simpangan baku.

$$\sigma = \sqrt{\frac{1}{MN-1} \sum_{X=1}^M \sum_{Y=1}^N (g(x, y) - \mu)^2} \quad \dots\dots\dots (2.5)$$

## 2.6. Teori Representasi Grafik

Dalam mode grafik, komputer menampilkan gambar dalam bentuk kumpulan titik yang tersusun dalam sebuah matriks atau array dua dimensi. Titik-titik tersebut disebut pixel, yang pada awalnya merupakan singkatan dari Picture Element (elemen gambar), tetapi akhirnya dinyatakan sebagai sebuah kata yang berdiri sendiri.

### 2.6.1. Representasi Bitmap

Representasi bitmap berarti sebuah gambar dibagi-bagi menjadi kotak-kotak kecil. Setiap kotak kecil (pixel) memiliki nilai (kecerahan atau warna) dan

lokasi masing-masing (David C. Kay: 1995). Setiap pixel yang ditampilkan pada layar monitor, dipetakan sebagai satu atau lebih bit dalam memori komputer. Karena itu, gambar yang ditampilkan dengan cara ini disebut sebagai bitmap yang artinya peta bit. Cara ini sering digunakan karena lebih mudah digunakan, tanpa batas, dan dapat berlaku untuk semua gambar.

Kelebihan representasi bitmap adalah kemudahannya untuk menampilkan gambar secara rinci dengan pola-pola yang kompleks atau gambar fotorealistik, yang tidak dapat dengan mudah direpresentasikan sebagai model matematika: garis, kurva, dan bidang.

### **2.6.2. *JPG/JPEG (Joint Photographic Experts Group)***

Tipe file JPG sangat sering digunakan untuk web atau blog. File JPG menggunakan teknik kompresi yang menyebabkan kualitas gambar turun (lossy compression). Setiap kali menyimpan ke tipe JPG dari tipe lain, ukuran gambar biasanya mengecil, tetapi kualitasnya turun dan tidak dapat dikembalikan lagi. Ukuran file BMP dapat turun menjadi sepersepuluhnya setelah dikonversi menjadi JPG. Meskipun dengan penurunan kualitas gambar, pada gambar-gambar tertentu (misalnya pemandangan), penurunan kualitas gambar hampir tidak terlihat mata.

### **2.6.3. *TIFF (Tagged Image File Format)***

Disingkat dengan TIFF (*Tagged Image File Format*) Suatu format file image yang standar untuk grafik bit-mapped yang beresolusi tinggi. File TIFF memiliki kualitas gambar yang cukup baik. Ada berbagai versi turunan dari TIFF. Format file *grafis bitmapped* TIFF yang umum dipakai dikembangkan oleh Aldus dan Microsoft untuk menyimpan file citra *monochrome*, *grayscale*, warna 8 dan 24 bit.

## **2.7. *Metode Global Image Matching***

Mewarnai gambar *grayscale* dapat dilakukan dengan beberapa cara, diantaranya adalah dengan mentransfer warna dari sebuah gambar warna ke gambar *grayscale*. Pentransferan warna secara keseluruhan dari gambar warna ke

gambar *grayscale* dapat dilakukan dengan mencocokkan tingkat kecerahan dan informasi tekstur diantara kedua gambar (Simon Premoze: 2002).

Secara umum, ada beberapa tahap sederhana yang harus dilakukan untuk mentransfer warna ke gambar *grayscale* dengan metode global image matching.

#### 1. Langkah I: Mengkonversi gambar ke $\alpha\beta$ *color space*.

Kedua gambar yang diinputkan, yaitu sebuah gambar warna dan sebuah gambar *grayscale*, harus dikonversi ke  $\alpha\beta$  *color space* sebelum proses selanjutnya dijalankan. Hal ini disebabkan karena  $\alpha\beta$  *color space* memiliki komponen L (*luminance*) yang dapat mempermudah proses pencocokan, dan juga komponen  $\alpha$  dan  $\beta$  yang dapat ditransfer dari gambar warna ke gambar *grayscale* setelah proses pencocokan dilakukan. Dengan mengasumsikan bahwa kedua gambar yang diinputkan adalah gambar RGB *color space*, maka proses yang perlu dilakukan adalah mengubah RGB *color space* menjadi  $\alpha\beta$  *color space*. Perlu diketahui, bahwa  $\alpha\beta$  *color space* di sini tidak sama dengan standar CIE LAB, keduanya hanya memiliki persamaan teori, yaitu memiliki sebuah komponen luminance (L) dan 2 buah komponen warna ( $\alpha$  dan  $\beta$ ).

Proses konversi dari RGB ke  $\alpha\beta$  ini memerlukan 2 tahap (Erik Reinhard: 2001, 34-41), yaitu:

##### a. Mengkonversi RGB ke LMS (*Long Middle Short wavelengths*).

LMS merupakan salah satu cara untuk merepresentasikan warna dengan 3 buah variabel, yaitu *long wavelength* (L), *middle wavelength* (M), dan *short wavelength* (S). LMS dan RGB merupakan dua buah cara yang hampir sama dalam merepresentasikan warna. LMS dipilih karena  $\alpha\beta$  merupakan transformasi dari LMS.

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \dots\dots\dots (2.6)$$

- b. Mengkonversi LMS ke  $l\alpha\beta$ .

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \dots\dots\dots (2.7)$$

2. Langkah II: Membuat dan menyesuaikan histogram tingkat kecerahan gambar.

Untuk mentransfer nilai kromatik (komponen  $\alpha$  dan  $\beta$ ) dari gambar warna ke gambar *grayscale*, setiap pixel pada gambar *grayscale* harus dicocokkan dengan pixel pada gambar warna. Pencocokan ini dilakukan dengan membandingkan tingkat kecerahan dan statistika pixel di sekeliling kedua pixel yang dibandingkan.

3. Langkah III: Menghitung statistika pixel.

Setelah melakukan proses luminance remapping, perlu dilakukan perhitungan statistika pixel untuk mencocokkan pixel pada gambar warna dengan pixel pada gambar *grayscale*.

Proses ini meliputi beberapa tahap, yaitu:

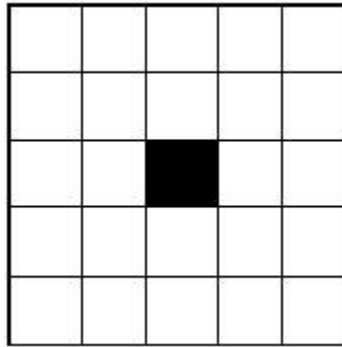
- a. Pemilihan sejumlah pixel secara acak dari gambar warna sebagai sampel.

Pembatasan jumlah pixel gambar warna dengan penggunaan sampel ini bertujuan untuk mengurangi jumlah perbandingan yang dilakukan oleh setiap pixel pada gambar *grayscale* sehingga mengurangi waktu perhitungan. Pada umumnya, jumlah sampel sebanyak 200 pixel cukup efektif dalam algoritma ini.

- b. Perhitungan statistika.

Perhitungan statistika gambar warna dilakukan terhadap pixel sampel yang telah diambil sebelumnya. Sedangkan perhitungan statistika gambar *grayscale* dilakukan terhadap setiap pixel. Adapun perhitungan statistika yang dilakukan adalah perhitungan rata-rata (*mean*) dan simpangan baku (standar deviasi) di

sekeliling pixel sesuai dengan ukuran yang diinputkan oleh pengguna. Untuk kebanyakan gambar, ukuran sekeliling pixel 5x5 bekerja dengan baik.



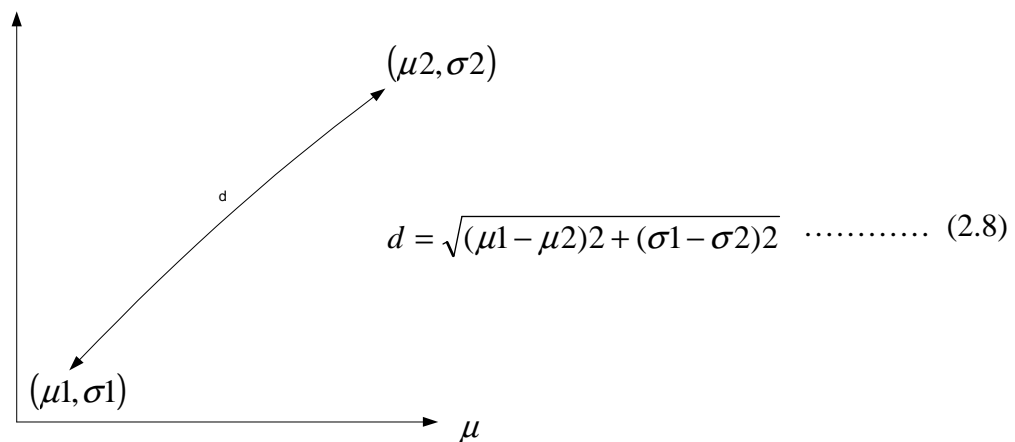
**Gambar 2.5. Ukuran sekeliling pixel 5x5**

4. Langkah IV: Mencocokkan pixel.

Setiap pixel pada gambar *grayscale* dicocokkan dengan pixel sampel yang telah diperoleh dari gambar warna. Menentukan pixel yang paling cocok antara gambar warna dan gambar *grayscale* dilakukan dengan menghitung bobot rata-rata dan simpangan baku tingkat kecerahan pixel di sekelilingnya.

Dengan menganggap rata-rata ( $\mu$ ) dan simpangan baku ( $\sigma$ ) tersebut sebagai koordinat, maka pixel gambar warna yang paling cocok adalah pixel yang memiliki jarak terpendek ( $d$ ) dengan pixel pada gambar *grayscale*.

Adapun rumus yang digunakan adalah:



langkah III dan langkah IV ini biasa disebut *texture synthesis* dengan sampling *local models*, yaitu mencari tekstur yang sama pada 2 gambar yang berbeda

dengan menghitung selisih (perbedaan distribusi) sekeliling kedua pixel yang dibandingkan. Selisih ini bernilai kecil jika kedua pixel yang dibandingkan mirip, dan bernilai besar jika keduanya berbeda jauh.

5. Langkah V: Mentransfer warna dan menghasilkan gambar yang berwarna.

Ketika ditemukan pixel yang paling cocok, nilai kromatik  $\alpha$  dan  $\beta$  ditransfer ke pixel target pada gambar *grayscale*, sementara tingkat kecerahan pixel tetap disimpan. Setelah proses pewarnaan di atas, maka gambar *grayscale* yang telah memiliki warna dalam  $\alpha\beta$  color space harus diubah kembali ke RGB color space untuk menampilkannya.

Langkah ini meliputi 2 proses (Erik Reinhard: 2001).

yaitu:

a. Mengkonversi  $\alpha\beta$  ke LMS.

Matriks transformasi yang digunakan adalah:

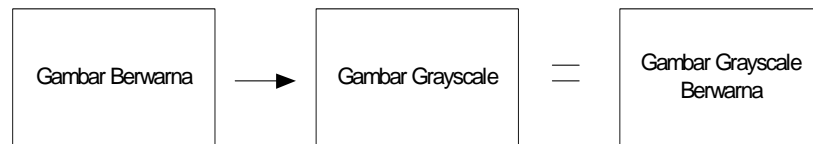
$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} L \\ \alpha \\ \beta \end{bmatrix} \dots\dots\dots (2.9)$$

b. Mengkonversi LMS ke RGB.

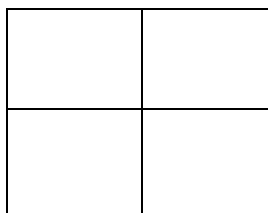
Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \dots\dots\dots (2.10)$$

### 2.7.1 Contoh Penerapan Metode *Global Image Matching*



Gambar yang memiliki ukuran pixel 2 x 2



#### 2.7.1.1 Gambar Berwarna

##### 1. Pengkonversian RGB ke $l\alpha\beta$ Untuk Gambar Berwarna

Untuk mengubah RGB menjadi  $l\alpha\beta$  memerlukan 2 tahap

- RGB ke LMS (*Long Middle Short Wavelengths*) untuk gambar berwarna.

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \dots\dots\dots (2.6)$$

Pixel 1

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 4 \end{bmatrix}$$

Pixel 2

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix}$$

Pixel 3

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 2 \end{bmatrix}$$

Pixel 4

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix}$$

- LMS ke  $\alpha\beta$  untuk gambar berwarna.

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \dots\dots\dots (2.7)$$

Pixel 1

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 3 \end{bmatrix}$$

Pixel 2

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix}$$

Pixel 3

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 3 \end{bmatrix}$$



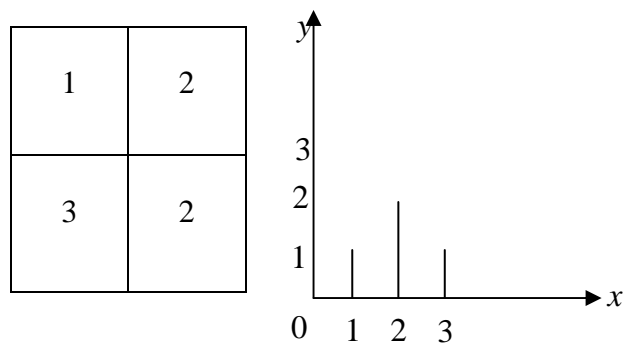
Pixel 4

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix}$$

## 2. Membuat Histogram Gambar berwarna

Berfungsi untuk menyesuaikan tingkat kecerahan gambar berwarna dan grayscale.

### Histogram gambar berwarna



## 3. Menghitung Statistika Pixel

Menggunakan rumus mencari rata-rata

$$\mu = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N g(x, y) \quad \dots\dots\dots (2.4)$$

$$\begin{aligned} \mu &= \frac{1}{M \times N} \\ &= \frac{1}{25} \sum_{x=1}^5 \sum_{y=1}^5 g(x, y) \\ &= \frac{1}{25} \times 100 \\ &= \frac{100}{25} = 4 \end{aligned}$$

Menggunakan Rumus mencari simpangan baku

$$\sigma = \sqrt{\frac{1}{MN-1} \sum_{x=1}^M \sum_{y=1}^N (g(x, y) - \mu)^2} \dots\dots\dots (2.5)$$

$$\sigma = \sqrt{\frac{1}{25-1} \sqrt{(100-4)}} = 1.5$$

### 2.7.1.2 Gambar Grayscale

#### 1. Pengkonversian RGB ke laβ Untuk Gambar Berwarna

Untuk mengubah RGB menjadi laβ memerlukan 2 tahap

- RGB ke LMS (*Long Middle Short Wavelengths*) untuk gambar berwarna.

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \dots\dots\dots (2.6)$$

Pixel 1

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 4 \end{bmatrix}$$

Pixel 2

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix}$$

Pixel 3

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 2 \end{bmatrix}$$

Pixel 4

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1567 & 0.5783 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix}$$

- LMS ke  $\alpha\beta$  untuk gambar berwarna.

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \dots\dots\dots (2.7)$$

Pixel 1

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 3 \end{bmatrix}$$

Pixel 2

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix}$$

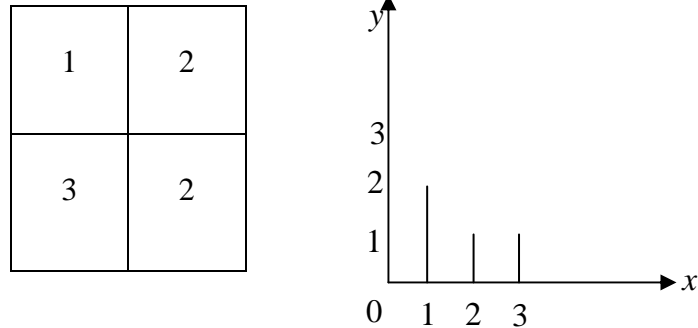
Pixel 3

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 3 \end{bmatrix}$$

Pixel 4

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix}$$

## 2. Histogram Gambar Grayscale



## 3. Menghitung Statistika Pixel

Menggunakan rumus mencari rata-rata

$$\mu = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N g(x, y) \quad \dots\dots\dots (2.4)$$

$$\begin{aligned}
 \mu &= \frac{1}{M \times N} \\
 &= \frac{1}{25} \sum_{x=1}^5 \sum_{y=1}^5 g(x, y) \\
 &= \frac{1}{25} \times 100 \\
 &= \frac{100}{25} = 4
 \end{aligned}$$

Menggunakan Rumus mencari simpangan baku

$$\sigma = \sqrt{\frac{1}{MN-1} \sum_{x=1}^M \sum_{y=1}^N (g(x, y) - \mu)^2} \quad \dots\dots\dots (2.5)$$

$$\sigma = \sqrt{\frac{1}{25-1}} \sqrt{(100-4)} = 1.5$$

### 2.7.1.3 Pencocokan Pixel

Menggunakan rumus standar deviasi

$$\begin{aligned}
 d &= \frac{\sqrt{(\mu_1 - \mu_2)^2 + (\sigma_1 - \sigma_2)^2}}{\sqrt{(\mu_1 - \mu_2)^2 + (\sigma_1 - \sigma_2)^2}} \dots\dots\dots (2.8) \\
 &= \frac{\sqrt{(4 - 1)^2 + (3 - 2)^2}}{\sqrt{(3)^2 + (1)^2}} \\
 &= \frac{\sqrt{10}}{5}
 \end{aligned}$$

#### 2.7.1.4 Proses Pentransferan

##### 1. Pembalikan laβ ke RGB

- Mengkonversi laβ ke LMS

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} L \\ \alpha \\ \beta \end{bmatrix} \dots\dots\dots (2.9)$$

Pixel 1

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 4 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 4 \end{bmatrix}$$

Pixel 2

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix}$$

Pixel 3

$$\text{Pixel 4} \quad \begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 4 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}$$

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 4 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ 4 \end{bmatrix}$$

- Mengkonversi LMS ke RGB

Matriks transformasi yang digunakan adalah:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \dots\dots\dots (2.10)$$

Pixel 1

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \\ 4 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix}$$

Pixel 2

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 3 \end{bmatrix}$$

Pixel 3

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix}$$

Pixel 4

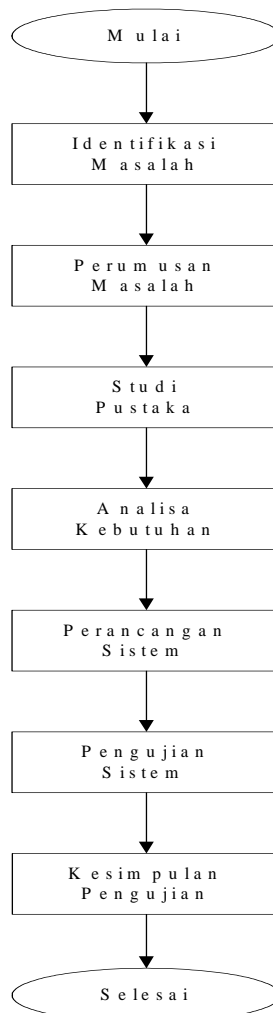
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} 4 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 4 \end{bmatrix}$$

## **BAB III**

### **METODOLOGI PENELITIAN**

Pada bab ini akan dipaparkan tentang langkah-langkah yang digunakan untuk membahas permasalahan yang diambil dalam penelitian. Pada bagian ini juga dijelaskan metode yang digunakan dalam proses pengembangan perangkat lunak.

Adapun alur pengerjaan tugas akhir ini digambarkan pada diagram alir berikut :



**Gambar 3.1 Diagram Alir Pelaksanaan Tugas Akhir**



### 3.1 Identifikasi Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya pada bab I, maka permasalahan yang diangkat pada tugas akhir ini adalah bagaimana membangun aplikasi untuk mentransfer warna dari citra yang berwarna ke citra yang tidak berwarna (*grayscale*) menggunakan metode global image matching.

### 3.2 Perumusan Masalah

Tahapan selanjutnya adalah merumuskan masalah yang akan diselesaikan dalam tugas akhir ini. Tujuan dari rumusan masalah adalah untuk mendapatkan point-point penting yang akan dibahas kemudian point-point masalah tersebut akan diselesaikan satu persatu untuk mencapai tujuan akhir yang telah ditetapkan sebelumnya.

### 3.3 Studi Pustaka

Studi pustaka dilakukan dengan cara :

a. Wawancara (*Interview*)

Wawancara dilakukan dengan menanyakan langsung kepada orang-orang yang ahli tentang pengolahan citra

b. Studi Pustaka (*Libery Research*)

Studi kepustakaan atau kajian pustaka dilakukan untuk mencari dan mempelajari serta mendalami informasi tentang pengolahan citra dan metode global image matching. Sumber kepustakaan diambil karya ilmiah yang berasal dari buku-buku maupun internet. Karya ilmiah yang dimaksud adalah berupa tulisan ilmiah yang berbentuk artikel, prosiding, buku, *e-book* (buku elektronik), dan lain-lain.

Studi pustaka bertujuan untuk mengumpulkan seluruh informasi yang mendukung penelitian pada tugas akhir ini.

### 3.4 Analisa Kebutuhan Aplikasi

Tahap analisa kebutuhan aplikasi dilakukan untuk mengetahui kebutuhan pengguna terhadap aplikasi yang akan dibuat. Hal ini perlu dilakukan agar aplikasi yang dikembangkan sesuai dengan kebutuhan pengguna.

### 3.4.1 Analisa Kebutuhan Data

a. Data masukan (*inputan*)

*Input* merupakan elemen dari sistem yang bertugas untuk menerima seluruh masukan data, dimana masukan tersebut dapat berupa jenis data, dan sebagainya.

b. Data proses

Proses merupakan elemen dari sistem yang bertugas untuk mengolah atau memproses seluruh masukan data menjadi suatu informasi yang lebih berguna.

c. Data keluaran (*output*)

*Output* merupakan hasil dari input yang telah diproses oleh bagian pengolah dan merupakan tujuan akhir sistem.

### 3.4.2 Analisa Pengguna

Yang menjadi sasaran pengguna dari aplikasi ini adalah semua individu pengguna yang ingin memperindah citra *grayscale* menjadi berwarna sehingga bisa memperindah citra yang diinginkan.

## 3.5 Perancangan Sistem

Pada tahapan ini dilakukan perancangan antarmuka (*input* dan *output*), perancangan proses dan merancang prosedur berdasarkan hasil dari analisa kebutuhan aplikasi yang telah dilakukan sebelumnya. Aplikasi yang dibuat tidak memerlukan media *database*. Penyimpanan hasil adalah pada komputer local (*Hard disk*). Pada tahap desain ini, *tool* yang digunakan adalah *Microsoft Office Visio 2003*. Hasil dari analisa kebutuhan aplikasi digambarkan kedalam bentuk diagram alir ( *Flow Chart*).

### **3.6 Pengujian dan Penerapan**

Sebelum perangkat lunak dapat digunakan, maka harus dilakukan pengujian terlebih dahulu. Pengujian difokuskan pada pencarian semua kemungkinan kesalahan, dan memeriksa apakah sistem telah sesuai dengan yang diinginkan.

Pengujian dilakukan bertujuan untuk mengetahui tingkat keberhasilan aplikasi dalam mentransfer warna dari citra yang berwarna ke citra grayscale.

### **3.7 Kesimpulan dan Saran**

Tahapan akhir dari penelitian adalah penarikan kesimpulan berdasarkan hasil yang telah diperoleh dari tahap sebelumnya, serta memberikan saran-saran serta untuk penyempurnaan dan pengembangan penelitian ini.

## **BAB IV**

### **ANALISIS DAN PERANCANGAN**

Analisis perangkat lunak merupakan langkah pemahaman persoalan sebelum mengambil tindakan atau keputusan dalam perancangan sistem yang akan dibuat. Sedangkan tahap perancangan sistem adalah tahapan yang dilakukan setelah tahap analisis yang mendefinisikan kebutuhan-kebutuhan fungsional dan mempersiapkan rancang bangun implementasi yang akan menggambarkan bagaimana sistem tersebut akan dibentuk.

#### **4.1 Hasil Analisis**

##### **4.1.1 Analisis Kebutuhan Sistem**

Setelah melakukan analisis, maka dapat diketahui kebutuhan sistem yang akan dibangun, seperti kebutuhan masukan sistem, kebutuhan keluaran sistem, kebutuhan fungsi dan antar muka sistem yang akan dibuat, serta kebutuhan perangkat lunak yang dibutuhkan dalam pembuatan aplikasi, sehingga sistem yang dibangun sesuai dengan yang diharapkan.

###### **4.1.1.1 Kebutuhan Masukan**

Data masukan yang dibutuhkan dalam pembuatan aplikasi ini adalah berupa:

1. Data gambar berwarna.
2. Data gambar *grayscale*.

###### **4.1.1.2 Kebutuhan Fungsi Atau Proses**

Proses yang berjalan pada aplikasi ini adalah :

1. Proses input data

Proses input data merupakan proses memasukkan data untuk disimpan kedalam local drive. Proses input data ini meliputi:

- a. Proses input data gambar berwarna
- b. Proses input data gambar grayscale

## 2. Proses pentransferan warna

Secara umum, proses pentransferan warna pada sebuah citra dapat dilihat pada gambar 4.1 dibawah ini.



Gambar 4.1 *Flowchart* Alur Proses Pentransferan Warna.

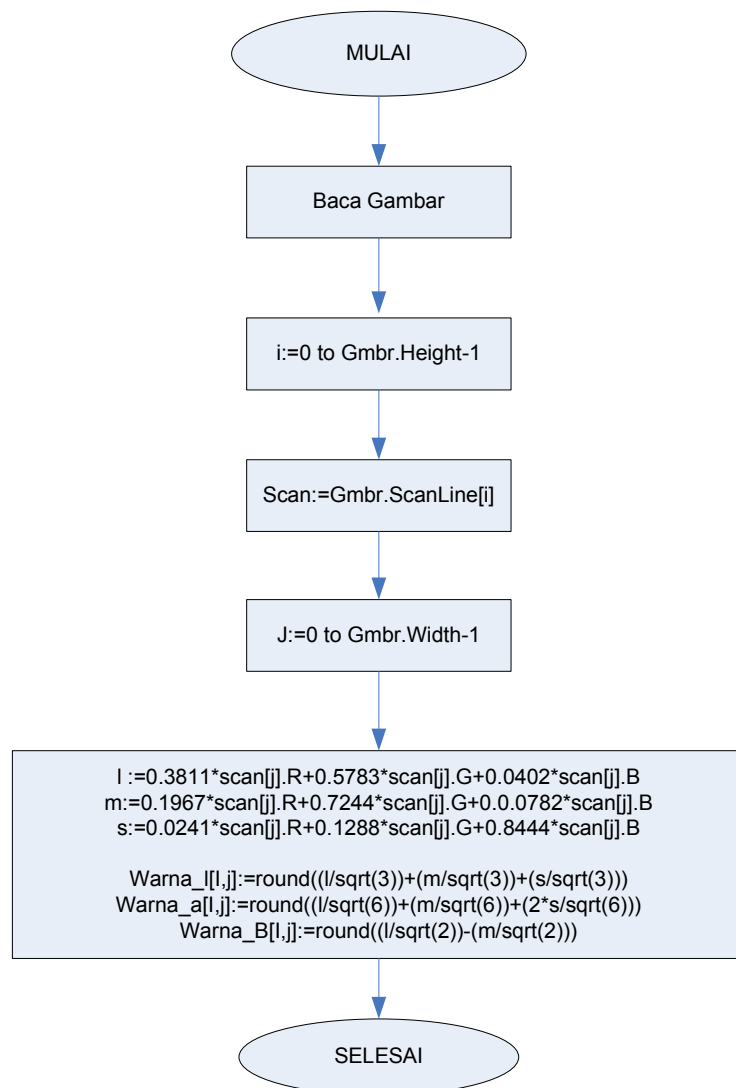
Proses pada gambar 4.1 diatas dapat dijelaskan sebagai berikut :

### a. Proses input gambar.

Proses penginputan gambar berwarna dan gambar *grayscale*, kedua gambar yang diinput harus memiliki ukuran pixel yang sama, kedua gambar yang diinput harus sebuah gambar orang berwarna dan gambar orang yang tidak berwarna (*grayscale*).

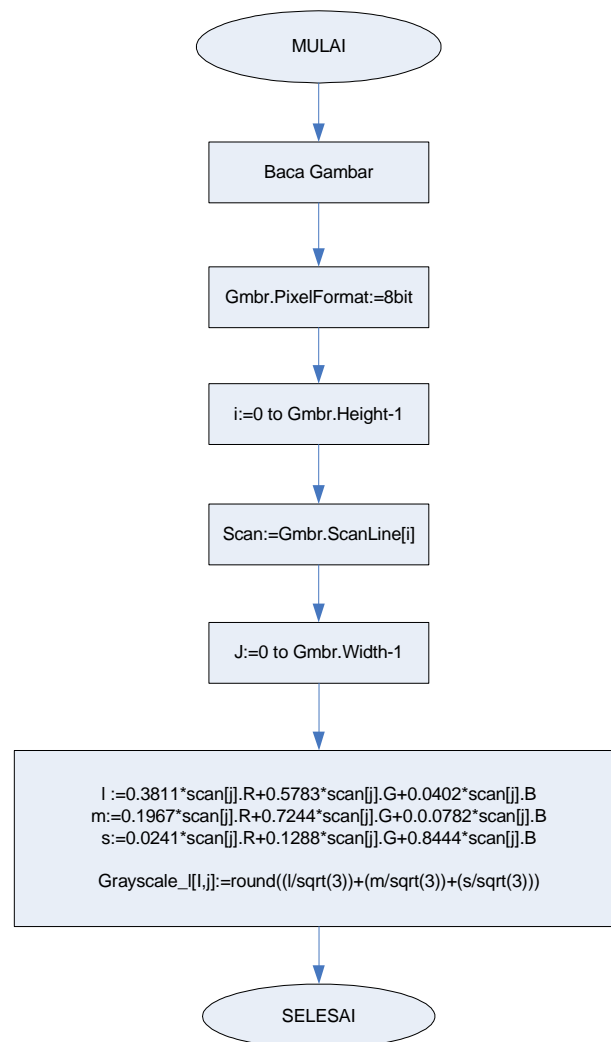
b. Proses konversi RGB ke  $l\alpha\beta$ .

Proses pengkonversian dari RGB ke  $l\alpha\beta$  dilakukan sebanyak 2 kali, yaitu pengkonversian untuk gambar berwarna dan pengkonversian untuk gambar *grayscale*, beberapa langkah yang dilakukan dalam proses pengkonversian pada gambar berwarna dapat dilihat pada gambar 4.2 dibawah ini.



Gambar 4.2 Flowchart Konversi RGB ke  $l\alpha\beta$  Gambar Berwarna.

Proses pentransferan pada gambar *grayscale* dapat dilihat pada gambar 4.3 dibawah ini.



Gambar 4.3 *Flowchart* Konversi RGB ke  $\alpha\beta$  Gambar *Grayscale*.

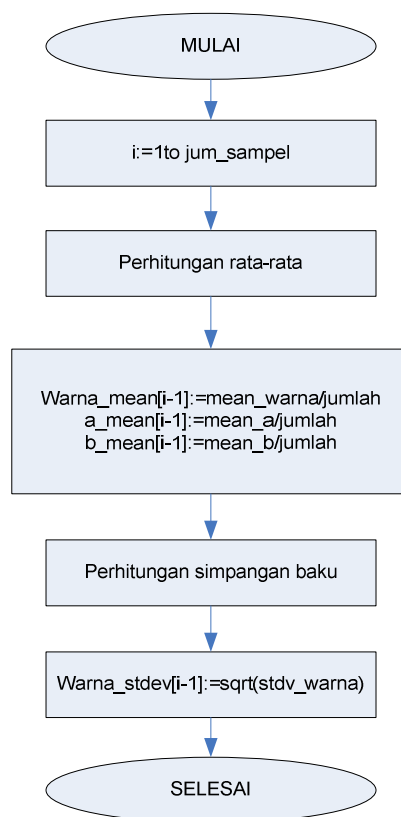
Pada aplikasi yang dibuat, proses konversi ini menghasilkan gambar berwarna dan gambar *grayscale* dalam format  $\alpha\beta$  yang disimpan dalam variable warna\_l, warna\_a, warna B, dan *grayscale*

- c. Perhitungan Rata-rata dan simpangan baku sekeliling pixel.

Perhitungan statistika (rata-rata dan simpangan baku) sekeliling pixel dilakukan sebanyak dua kali, yaitu untuk

gambar berwarna yang telah disesuaikan dan untuk gambar grayscale. Ukuran sekeliling pixel yang telah diperhitungkan adalah sesuai dengan input pengguna, ukuran sekeliling pixel adalah 5x5.

Proses perhitungan dapat dilihat pada gambar 4.4 dibawah ini.

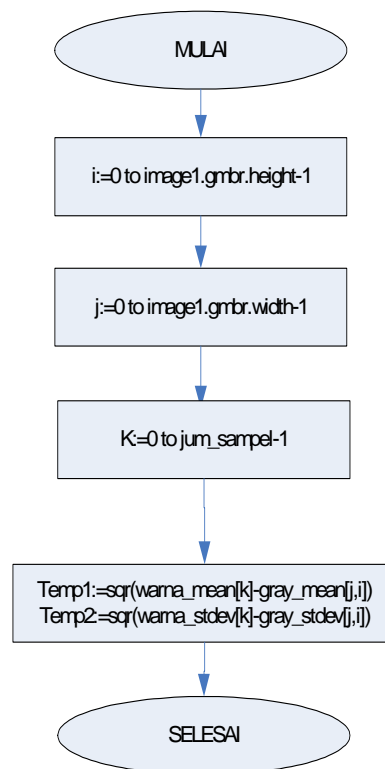


Gambar 4.4 *Flowchart* Perhitungan Rata-rata dan simpangan baku.



d. Pencocokan Pixel.

Proses ini dilakukan dengan mencari jarak terpendek antara titik rata-rata dan simpangan baku gambar warna dengan gambar *grayscale*. Proses ini pencocokan dapat dilihat pada gambar 4.5 dibawah ini.

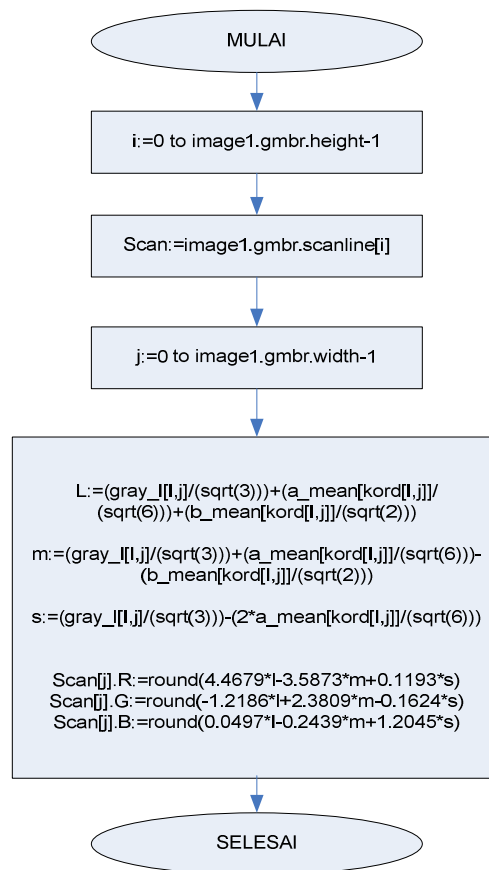


Gambar 4.5 *Flowchart* Pencocokan pixel.

e. Pentransferan Warna

Proses pentransferan warna merupakan proses terakhir dalam serangkaian proses pewarnaan. Proses ini bertujuan memberi warna (nilai kromatik  $\alpha$  dan  $\beta$ ) ke gambar grayscale dan menampilkan gambar tersebut dalam format RGB. Untuk itu perlu dilakukan proses untuk mengkonversi  $\alpha\beta$  ke RGB.

Lebih rinci proses tersebut dapat dilihat pada gambar 4.6 dibawah ini.



Gambar 4.6 *Flowchart* Pentransferan Warna.

### 3. Proses pencetakan data

Proses pencetakan data merupakan pencetakan terhadap data kedalam bentuk *hard copy*.

#### 4.1.1.3 Kebutuhan Keluaran

Data keluaran dari aplikasi ini berupa :

Gambar *grayscale* yang sudah menjalani proses pentransferan warna dengan hasil menjadi berwarna.

#### 4.1.1.4 Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam pembuatan Tugas Akhir ini adalah :

1. *Microsoft Office Visio 2003* digunakan dalam pembuatan *Flowchart system*.
2. Bahasa Pemograman *Borland Delphi 7.0* digunakan dalam proses pembuatan aplikasi.

## 4.2 Perancangan Sistem

Berdasarkan pada hasil analisis yang telah dilakukan sebelumnya, maka dapat dilakukan perancangan terhadap sistem yang akan dibangun.

### 4.2.1 Deskripsi Umum Sistem

Aplikasi Pentransferan warna merupakan aplikasi yang membantu melakukan proses pentranferan warna pada gambar berwarna ke gambar *grayscale*. Dengan adanya aplikasi ini diharapkan mempermudah pengguna yang ingin memperindah suatu citra grayscale dengan nuansa warna dari suatu gambar berwarna. Sehingga kita bisa memperindah koleksi photo tua hitam putih, photo-photo yang sudah kusam, dan photo-photo hasil dari scanner.

### 4.2.2 Perancangan Antar Muka

Perancangan antar muka merupakan perancangan bentuk tampilan dari aplikasi yang nantinya dapat menjadi navigasi bagi pengguna dalam menggunakan aplikasi. Antar muka pengguna dirancang dengan berbasis GUI (*Graphical User Interface*) agar pengguna merasa nyaman dan mudah dalam menggunakan aplikasi.

#### 4.2.2.1 Tampilan Form Utama

The image shows a software interface for 'GLOBAL IMAGE PROCESSING'. It features a standard Windows-style window with a title bar and a menu bar. The main workspace is divided into two identical vertical panels. Each panel contains a column of four buttons: 'BUKA FILE', 'KONVERSI', 'HISTOGRAM', and 'RATA-RATA'. To the right of these buttons is a large, empty rectangular box, likely for displaying image results. At the bottom of the window, there is a section labeled 'PROSES' followed by a long, empty text input field.

Gambar 4.7 Tampilan Form Utama

Form utama merupakan form inti dari aplikasi. Form ini berisi navigasi-navigasi yang menghubungkan pengguna dengan form-form lain yang ada pada aplikasi ini.

Pada form ini berisi menu-menu yang akan dijalankan dalam proses pentransferan.

1. Tombol input gambar  
Digunakan untuk memasukkan gambar yang ingin digunakan
2. Tombol Konversi  
Digunakan untuk melanjutkan ke proses pengkonversian
3. Tombol Histogram  
Digunakan melihat histogram pada citra yang diinputkan.
4. Tombol Tombol Pencocokan Pixel  
Digunakan untuk melanjutkan proses pencocokan pixel sebelum pentranferan warna dilakukan
5. Tombol Pentranferan Warna  
Digunakan untuk menyelesaikan proses pentransferan warna.

#### 4.2.2.2 Tampilan Form Histogram gambar berwarna

HISTOGRAM	
GAMBAR HISTOGRAM	NILAI HISTOGRAM

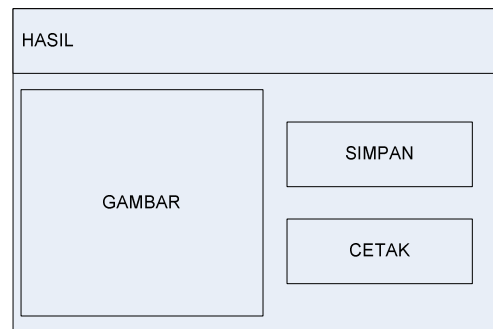
Gambar 4.8 Tampilan Form Histogram gambar berwarna

#### 4.2.2.3 Tampilan Form Histogram gambar *grayscale*

HISTOGRAM	
GAMBAR HISTOGRAM	NILAI HISTOGRAM

Gambar 4.9 Tampilan Form Histogram gambar *grayscale*

#### 4.2.2.4 Tampilan Hasil Pentransferan



Gambar 4.10 Tampilan Hasil Pentransferan

## **BAB V**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **5.1 Implementasi Sistem**

Implementasi merupakan tahap dimana sistem siap untuk dioperasikan, hal ini dilakukan setelah perangkat lunak selesai dikerjakan. Pada tahap implementasi sistem ini, diharapkan sistem yang telah dirancang siap untuk dioperasikan pada keadaan yang sebenarnya, sehingga akan diketahui apakah sistem yang dibuat benar-benar dapat menghasilkan tujuan yang diinginkan dan sesuai dengan tujuan yang diharapkan.

##### **5.1.1 Lingkungan Implementasi**

Lingkungan Implementasi sistem ada 2 yaitu: lingkungan perangkat keras dan lingkungan perangkat lunak.

###### **1. Perangkat Keras Komputer**

Perangkat keras komputer yang digunakan mempunyai spesifikasi sebagai berikut:

- a. *Processor Dual Core*
- b. *Memory* 1024 GB
- c. *Hard disk* berkapasitas 80 GB

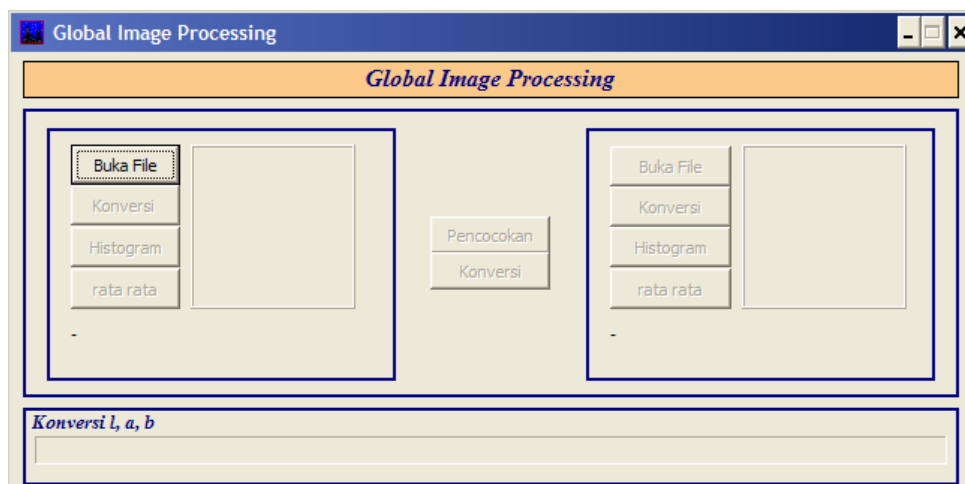
###### **2. Perangkat Lunak**

Perangkat lunak dalam implementasi ini menggunakan:

- a. Sistem Operasi Windows XP.
- b. Borland Delphi 7

## 5.1.2 Aplikasi Pentransferan Warna Pada Gambar Grayscale Menggunakan Metode Global Image Matching

### 5.1.2.1. Menu Utama



Gambar 5.1 Tampilan Menu Utama

Ada beberapa sub menu yang terdapat pada menu utama, diantaranya:

1. Tombol Buka File  
Digunakan untuk memasukkan gambar yang ingin digunakan
2. Tombol Konversi  
Digunakan untuk melanjutkan ke proses pengkonversian
3. Tombol Histogram  
Digunakan melanjutkan proses histogram dan melihat histogram pada citra yang diinputkan.
4. Tombol Tombol Pencocokan Pixel  
Digunakan untuk melanjutkan proses pencocokan pixel sebelum pentranferan warna dilakukan
5. Tombol Pentransferan Warna  
Digunakan untuk menyelesaikan proses pentransferan warna.



## **5.2 Pengujian Sistem**

Setelah tahap implementasi dilakukan maka dilanjutkan dengan pengujian dari implementasi yang telah dibuat. Tahap pengujian diperlukan agar dapat diketahui hasil dari program implementasi sistem. Pengujian dikatakan berhasil apabila warna yang ditransfer sesuai dengan aspek realita, sedangkan pengujian dikatakan gagal apabila warna yang ditransfer tidak sesuai dengan aspek realita.

### **5.2.1 Lingkungan Pengujian Sistem**

Pengujian sistem ini dilakukan pada lingkungan perangkat lunak dan perangkat keras sesuai dengan lingkungan implementasi.

### **5.2.2 Pengujian**

1. Pengujian Aplikasi Pentransferan Warna Pada Citra Digital menggunakan Metode Global Image Matching,
2. Pengujian dilakukan terhadap gambar-gambar yang memiliki tema yang sama.
3. Pengujian dilakukan terhadap gambar-gambar yang memiliki tema yang tidak sama.
4. Pengujian dilakukan terhadap objek yang sama.
5. Pengujian dilakukan sebanyak 3 kali untuk masing-masing gambar.

### **5.2.3 Hasil Pengujian**

Dari pengujian sistem yang telah dilakukan, sistem dapat memberikan hasil yaitu sebuah gambar grayscale yang telah memiliki warna. Hasil pengujian dapat dilihat pada *lampiran C*

### **5.2.4 Kesimpulan Pengujian**

Berdasarkan hasil pengujian sistem dalam pentransferan warna menggunakan metode *Global Image Matching*, dapat ditarik kesimpulan sebagai berikut:

1. Keluaran yang dihasilkan oleh sistem adalah sebuah gambar grayscale yang sudah ditransfer warna sehingga menjadi berwarna.
2. Metode *Global Image Matching* dapat digunakan dalam proses pentransferan warna pada citra atau gambar.
3. Kemampuan aplikasi dengan menggunakan metode *global image matching* ini memandang tema yang sama dalam proses pentransferan.
4. Warna yang dihasilkan dari proses pentransferan dipengaruhi oleh gambar berwarna yang dipilih, yakni warna mayoritas yang terdapat pada gambar berwarna.
5. Hasil dari proses pentransferan pada gambar memiliki hasil yang sama meskipun proses pengujian dilakukan lebih dari satu kali.

## **BAB VI**

### **PENUTUP**

#### **6.1 Kesimpulan**

Kesimpulan dari penelitian tugas akhir ini adalah sebagai berikut:

1. Warna yang dihasilkan dari proses pentransferan dipengaruhi oleh gambar berwarna yang dipilih, yakni warna mayoritas yang terdapat pada gambar berwarna, dan menggunakan tema yang sama.
2. Metode *Global Image Matching* dapat digunakan dalam proses pentransferan warna pada citra atau gambar.
3. Hasil dari pengujian menggunakan 55 citra yang berwarna diperoleh tingkat keberhasilan sebesar 90 %.
4. Hasil dari proses pentransferan pada gambar memiliki hasil yang sama meskipun proses pengujian dilakukan lebih dari satu kali.

#### **6.2 Saran**

Beberapa hal yang dapat diungkapkan sebagai saran untuk perbaikan di masa yang akan datang mengenai pentransferan warna dengan menggunakan metode *Global Image Matching* ini adalah sebagai berikut:

1. Menemukan beberapa algoritma pentransferan warna lainnya, sehingga aplikasi dapat menghasilkan ketepatan gambar yang lebih baik.
2. Menambah jenis format gambar yang dapat diproses dalam aplikasi ini, misalnya *png*, *gif*, dan lain sebagainya.
3. Memperbesar ukuran pixel gambar.

## DAFTAR PUSTAKA

- Crane, Randy. "A Simplified Approach to Image Processing". Prentice-Hall, 1997.
- Gose, Earl, Richard Johnsonbaugh, and Steve Jost. "Pattern Recognition and Image Analysis". Prentice Hall PTR, 1996.
- Kadir, Abdul, "*Pengenalan Sistem Informasi*", Andi Yogyakarta, 2003
- Loy, Gareth. "Colour Imaging." An Introduction to Computer Vision. 2002.
- Mulyanta, Edi S., *Dari Teori Hingga Praktik : "Pengolahan Digital Image dengan Photoshop CS "*.CV. Andi Offset (Penerbit Andi) : Yogyakarta, 2007.
- Rinaldi, "Pengolahan Citra Digital", halaman 2, [Online] Available :  
[http://informatika.org/~rinaldi/Buku/Pengolahan%20Citra%20Digital/Bab-1\\_Pengantar%20Pengolahan%20Citra.pdf](http://informatika.org/~rinaldi/Buku/Pengolahan%20Citra%20Digital/Bab-1_Pengantar%20Pengolahan%20Citra.pdf), diakses September 2008
- Tharom, Tabratas., "*Pengolahan Citra Pada Mobil Robot*", Penerbit ITB : Bandung, 2000.
- Wahyono, Teguh., *Sistem Informasi (Konsep Dasar, Analisis Desain dan Implementasi)*, Graha Ilmu : Yogyakarta, 2004.
- Welsh, Tomihisa, Michael Ashikhmin, and Klaus Mueller. "Transferring Color to Grayscale Image." Acm Transactions on Graphics. 2002.
- Wijaya, Marvin ,Ch., Priyono, Agus., *Pengolahan Citra Dijital Menggunakan MATLAB*, Penerbit Informatika : Bandung, 2007.